

MiniTerminal (BELC 1001)

Ansteuerung mit Festtextspeicher für Character LCD-Module

1 Allgemeines

- Mit den MiniTerminals BELC1001 können alle gängigen LCD-Character-Module über eine V24-Schnittstelle angesteuert werden.
- Das MiniTerminal BELC1001 verfügt standardmässig über einen Festtextspeicher von 2 kByte (je nach Typ aufrüstbar auf 6 kByte).
- Die Ansteuerung des MiniTerminals erfolgt über ESC-Sequenzen, die an den VT100 Standard angelehnt sind.
- Die Festtexte können über die serielle Schnittstelle und/oder über 7 TTL-Leitungen (J4) abgerufen werden.
- Die Ausgabe von Zeichen auf dem Display ist unabhängig von der Festtextausgabe und kann mit dieser beliebig gemischt werden (siehe Demo-Programm).
- Alle MiniTerminals der BELC1001- Reihe sind mechanisch kompatibel zu den von uns angebotenen Displays
- Die Versorgung erfolgt über **stabilisierte +5 Volt** an J6.

2 Datenübertragung

Die Datenübertragung erfolgt über die RS232 Schnittstelle des MiniTerminals.

Standardeinstellung des MiniTerminals: 9600 Baud / 8 Data-Bits / no parity / 2 Stop-Bits

Einstellungsmöglichkeiten: 1200, 2400, 4800, 9600 Baud

Die serielle Schnittstelle kann **nicht** gleichzeitig senden und empfangen.

Sie müssen deshalb das CTS-Signal beachten, wenn Sie die Tastatur verwenden.

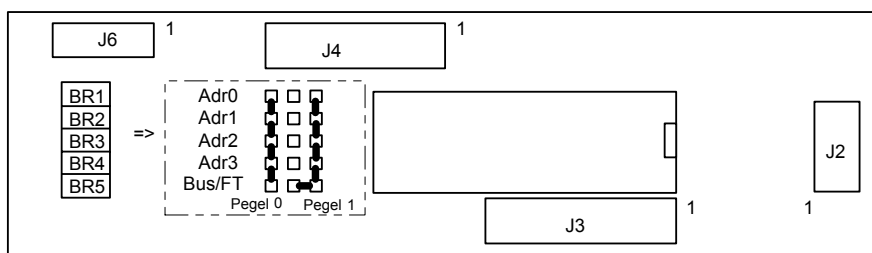
3 Adressierung

Standardeinstellung des MiniTerminals: Alle Adressen

Durch das Kodieren der Brücken BR1...BR4 kann eine Adressierung der MiniTerminals erfolgen.

Die Adressierung erfolgt binär, d. h. es können maximal 16 (Adr.0....15) MiniTerminals über eine Schnittstelle angesprochen werden.

Zu beachten: Das CTS und TXD Signal des MiniTerminals darf in diesem Fall nicht angeschlossen werden. Das MiniTerminal funktioniert in diesem Modus nur als passive Textanzeige mit der Festtexte aufgerufen, jedoch nicht programmiert werden können.



Pin 'Bus/FT' ist für 'automatischen Festtext' konfiguriert, d.h. auf Pegel 1 gelegt

4 Tastatur

Die Matrix-Tastatur kann aus maximal 8 Spalten und 4 Zeilen (32 Tasten) bestehen. Sie wird am Stecker J3 angeschlossen.

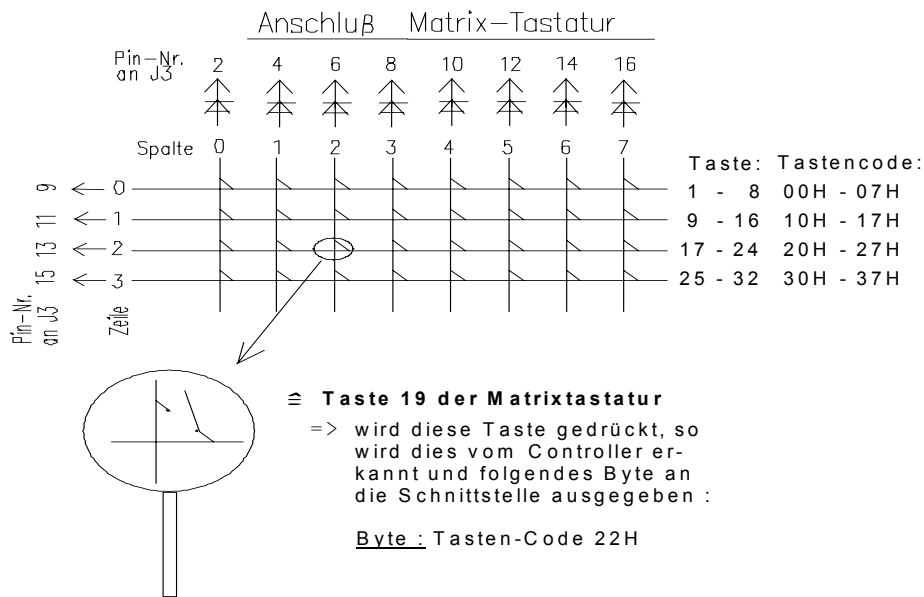
Wird eine Taste gedrückt, setzt das MiniTerminal die CTS-Leitung und sendet den Tasten-Code.

Der Tasten-Code wird an die serielle Schnittstelle weitergeleitet. Dort kann der Code durch ein Programm ausgelesen und beliebig weiterverarbeitet werden.

Bei gleichzeitigem Drücken von mehreren Tasten können die Tastaturausgangports des µP kurzgeschlossen werden.

Dies kann Fehlfunktionen des µP verursachen.

Aus diesem Grund sollte man an jedem Spalteneingang der Matrix-Tastatur eine Sperrdiode vorsehen (siehe Abb.).



5 Betriebsarten

Die Betriebsartenauswahl erfolgt im spannungslosen Zustand des MiniTerminals. Das Einlesen erfolgt nach Anlegen der Versorgungsspannung.

Grundsätzlich sind 2 Betriebsarten möglich:

- 1) BUS
- 2) Auto Festtext

zu 1) Betriebsart BUS (Vorwahl: BR5 = low (0))

In dieser Betriebsart erfolgen sämtliche Steuerbefehle und Festtextaufrufe nur über die serielle Schnittstelle.

Der Steckverbinder J4 ist hier deaktiviert. Es können somit keine Festtexte über die Hardwareseite aufgerufen werden.

zu 2) Betriebsart Auto Festtext (Vorwahl: BR5 = high(1))

In dieser Betriebsart können die gespeicherten Festtexte **zusätzlich** über die Pins FT0...FT6 am J4 aufgerufen werden.

Bitte beachten: J4 ist low activ. Bei offenen Anschlüssen sind alle Pins „high“ (1), d.h. Festtext 127 (Binär: 1111111) ist ausgewählt.

Siehe auch: ESQ- Befehl : Festtext ausgeben. (ESC D)

6 Inbetriebnahme des MiniTerminals

Bitte beachten Sie hierbei folgende Punkte. Nur so ist eine sichere Inbetriebnahme gewährleistet :

- Brücke BR5 in Stellung 'Bus' bringen (d.h. Pin 'FT/Bus' mit Masse verbinden).
- Schließen Sie das MiniTerminal gemäß Anschlußbelegung auf Seite 10 an.
- Verbinden Sie den angefertigten Stecker J6 mit dem Anschluß J6 auf Ihrer Platine. Achten Sie darauf, dass Pin 1 auf der entsprechend gekennzeichneten Seite angeschlossen wird. Stecken Sie dann die 9 pol. Buchse auf die serielle Schnittstelle (COM1 oder 2) an Ihrem PC.
- Legen Sie jetzt die +5 Volt Versorgungsspannung am Stecker J6 an.
- Die Konfigurationsdaten erscheinen auf dem Display. Falls das nicht der Fall ist, muß an dem Regler P1 auf der Platine der Kontrast so lange verändert werden bis die Daten gut zu sehen sind.

Konfigurationsdaten: **V**<Softwareversion> **T**<Displaytyp> **B**<Baudrate>
 Beispiel (Grundeinstellung): **V 1.5 T2 B0**

Hinweis:
 Display-Typ siehe 'ESC & Sn'
 Baudrate siehe 'ESC & Bn'

- Mit den Kommandos entsprechend der Beschreibung (siehe 'Escape-Sequenzen') oder mit Hilfe des Demo-Programms können Festtexte gespeichert und angezeigt werden.

Erweiterung :

Möchten Sie jetzt zusätzlich zur serielle Schnittstelle die Festtexte über die 7 TTL-Leitungen des Anschlusses J4 abrufen, dann müssen Sie

- Die Brücke BR5 in Stellung 'autom. Festtext' bringen (d.h. Pin'FT/Bus' mit +5 Volt verbinden)
- Den Stecker J4 entsprechend der Beschreibung 'Ansteuerung Festtextauswahl' (Kap. 8) anschließen. Beachten Sie dabei, daß am Pin 8('FT/Bus') von J4 schon +5 Volt anliegen, da dieser Anschluß mit Pin 'FT/Bus' (BR5) auf dem MiniTerminal verbunden ist.

Mit den 7 Steuerleitungen (FT0- FT6) können bis max. 128 Festtexte binär abgerufen werden. Ein nicht angeschlossener Pin wird als logisch high (1) interpretiert. (FT0...FT6 sind low aktiv)

Bsp.

FT0	FT1	FT2	FT3	FT4	FT5	FT6	Festtext Nr.
1	1	1	0	0	0	0	7
0	0	0	0	0	0	1	64
1	1	1	1	1	1	1	127

7 Beschreibung der Escape-Sequenzen

CR Wagen-Rücklauf / Neue Zeile 0d (hex) / 13 (dezimal)
ESC 1B (hex) / 27 (dezimal)

ESC [(Zeile) ; (Spalte) H | Cursor setzen

Zeile 0.. 3 / Spalte 0..39 Alle Zahlen ohne führende Nullen !

Beispiel: **PRINT #1,CHR\$(27);"[";CHR\$(2);";" ;CHR\$(1);CHR\$(9);"H";**
 setzt den Cursor in die 20. Spalte der 3. Zeile
 (Beachten: 2-stellige Spaltenzahlen auch getrennt ausgeben)

ESC & # | Display löschen / Cursor Home

Beispiel: **PRINT #1,CHR\$(27);"&#";**

ESC & C | Cursor ein

Beispiel: **PRINT #1,CHR\$(27);"&C";** schaltet den Cursor ein

ESC & D | Cursor aus

Beispiel: **PRINT #1,CHR\$(27);"&D";** schaltet den Cursor aus

ESC [K | Bis zum Ende der Zeile löschen

Beispiel: **PRINT #1,CHR\$(27);"[K";** löscht alle Zeichen bis zum Ende der aktuellen Zeile

ESC & S n | Display-Type auswählen

<i>n</i> =	0 (^@)	16 x 1 / 8 x 2
	1 (^A)	16 x 2
	2 (^B)	16 x 4
	3 (^C)	20 x 1
	4 (^D)	20 x 2
	5 (^E)	20 x 4
	6 (^F)	24 x 1
	7 (^G)	24 x 2
	8 (^H)	32 x 1
	9 (^I)	32 x 2
	10 (^J)	40 x 1
	11 (^K)	40 x 2
	12 (^L)	40 x 4

Beispiel: **PRINT #1,CHR\$(27);"&S";CHR\$(2);** Initialisiert das MiniTerminal für ein 16 x 4 Display
 Der ausgewählte Display-Typ wird im EEPROM abgespeichert

ESC & A adresse | Adresse vorgeben

ESC & A Z | alle selektiert (Default-Wert)

adresse = 0...15

Beispiel : **Print #1, CHR\$(27); "&A";CHR\$(1)** => nur MiniTerminal mit zugeordneter Adresse 1 wertet Daten aus
Print #1, CHR\$(27); "&AZ"; => alle angeschlossenen MiniTerminals werten Daten aus (Grundeinstellung)

ESC & Bn | Einstellen der Baudrate

n = 0 (9600), 1 (4800), 2 (2400), 3 (1200)

Beispiel: **PRINT #1,CHR\$(27);"&B";CHR\$(2);**
setzt die Baudrate auf 2400 Bits pro Sekunde

Diese Einstellung wird im EEPROM permanent abgespeichert.

7.1 Zeichen umdefinieren

ESC & L CG Zeichen 8x Muster | bestehendes Zeichen umdefinieren

CG = Zu verwendende CG-RAM-Adresse. Es stehen 8 Adressen zur Zeichenumdefinierung zur Verfügung. CG = Zahl von 1 - 8

Zeichen = Zeichen, welches umdefiniert werden soll. Hier kann ein beliebiges Zeichen des Character-Fonts angegeben werden

8x Muster = Hier muß das Bit-Muster des neuen Zeichens angegeben werden. Ein Zeichen besteht aus einer '5 x 7 Punkt-Matrix'. Nur 'Bit0 - Bit4' der 7 anzugebenden Bytes sind relevant. Als 8. Byte wird immer 'CHR\$(0)' angegeben.

Beispiel-Proramm (Qbasic) :

```

^Schnittstelle definieren
OPEN "COM1:9600,N,8,2,cs0,ds0,cd0,rs" AS 1
'Zeichen-Muster
pattern$ = CHR$(85)+CHR$(170)+CHR$(85)+CHR$(170)+CHR$(85)+CHR$(170)+CHR$(85)+CHR$(0)
INPUT "Umzudefinierendes Zeichen :"; zch$
INPUT "Zu verwendende CG-RAM-Adresse (1-8) :"; CGAdr%
'ESC-Sequenz zur Zeichenumdefinierung
PRINT #1, CHR$(27) + "&L";
PRINT #1, CHR$(CGAdr% - 1);
strg$ = zch$ + pattern$
l = LEN(strg$)
IF l <> 0 THEN
  FOR n = 1 TO l
    PRINT #1, MID$(strg$, n, 1);
    R$ = INPUT$(1, #1)
    IF R$ <> MID$(strg$, n, 1) THEN
      PRINT "Fehler beim Speichern": BEEP
      EXIT FOR
    END IF
  NEXT n
END IF

```

Erläuterung :

5x7 Matrix

0	1	0	1	0	1	0	1	0	1	-> CHR\$(85)
1	0	1	0	1	0	1	0	1	0	-> CHR\$(170)
0	1	0	1	0	1	0	1	0	1	
1	0	1	0	1	0	1	0	1	0	
0	1	0	1	0	1	0	1	0	1	
1	0	1	0	1	0	1	0	1	0	
0	1	0	1	0	1	0	1	0	1	
0	0	0	0	0	0	0	0	0	0	

Löschen von umdefinierten Zeichen:

Für umzudefinierendes Zeichen (zch\$) einen Wert zwischen CHR\$(10) - CHR\$(18) eingeben (nicht darstellbares Zeichen) und zu löschende CG-RAM-Adresse (CGAdr%) angeben.

7.2 Tastencode ändern

ESC & u Byte0 Byte31

Belegung des Tasten-Codes ändern

Beispiel: **PRINT #1,CHR\$(27);"&u";**
TEXT\$ = "abcdefghijklmnopqrstuvwxy,,:;?" ' neuer Tasten-Code
L = LEN(TEXT\$)
FOR I=1 TO L
PRINT #1,MID\$(TEXT\$,I,1);
R\$ = INPUT\$(1,#1)
IF R\$ <> MID\$(TEXT\$,I,1) THEN GOTO fehler ' Fehler beim Speichern
NEXT I
fehler:
PRINT "Fehler beim Speichern"
ende:
END

Der originale Tasten-Code wird der Reihe nach, durch die dem "u" folgenden 32 Byte ersetzt.
Der originale Tasten-Codes sieht folgendermaßen aus:

00H - 07H -> Taste 1 - 8
10H - 17H -> Taste 9 - 16
20H - 27H -> Taste 17 - 24
30H - 37H -> Taste 25 - 32

Die Tasten-Codes werden wie die Festtexte, ebenfalls im EEPROM permanent abgespeichert.

zu Beachten :

Die Einstellungen der Baudrate und der Umkodiertabelle bleiben auch bei Spannungsausfall gespeichert. Diese Einstellungen werden nur durch die o.g. Escape-Sequenzen geändert

7.3 Festtexte

Die Festtexte sind mit 0 .. 127 durchnummeriert.
Vor der Ausgabe eines Festtextes wird das Display gelöscht.
Ein Festtext darf das Sonderzeichen CR enthalten.

Das MiniTerminal kann mit maximal 3 EEPROMs à 2kByte bestückt werden.

Festtexte auslesen :

ESC W

Festtexte über die serielle Schnittstelle ausgeben

Beispiel: **PRINT #1,CHR\$(27);"W";**

Das MiniTerminal gibt alle gespeicherten Festtexte der Reihe nach über die serielle Schnittstelle aus. Jeder Festtext wird durch 'CHR\$(0)' abgeschlossen. Die Übertragung wird durch 2x 'CHR\$(0)' abgeschlossen (siehe Speicherformat unter 'Festtext laden').

Festtext laden :

ESC R	Festtexte über ser. Schnittstelle laden und im EEPROM abspeichern
--------------	---

Das MiniTerminal schickt jedes Zeichen nach dem Programmieren zurück. Wurde beim Programmieren ein Fehler festgestellt (z.B. EEPROM defekt oder Speicherplatz erschöpft), wird das Zeichen invertiert. (Fehlerüberprüfung siehe Demo-Programm Zeile 230-250.)

Das Speichern der Texte dauert ungefähr 1 Sec pro 100 Zeichen.

Beispiel:	PRINT #1,CHR\$(27);"R";	
	PRINT #1,"T";	1. Zeichen("T") des 1. Festtextes
	INPUT\$(1,#1)	warten, bis das Zeichen zurückgeschickt wird
	
	PRINT #1,CHR\$(0);	Ende des 1. Festtextes
	PRINT #1,"K";	1. Zeichen("K") des 2. Festtextes
	INPUT\$(1,#1)	warten, bis das Zeichen zurückgeschickt wird
	
	PRINT #1,CHR\$(0);	Ende des n. Festtextes
	PRINT #1,CHR\$(0);	Ende der Übertragung

Achtung: Alle im EEPROM gespeicherten Festtexte werden bei der Übertragung überschrieben.

Geeignete Vorgehensweise bei Abänderung oder Erweiterung bestehender Festtexte:

1. Bestehende Festtexte mit Befehl 'ESC W' auslesen und in Datei abspeichern (pro Zeile 1 Festtext)
2. Festtexte in Datei mit ASCII-Editor bearbeiten (Speicherformat beachten)
3. Festtexte aus Datei zeilenweise einlesen und mit Befehl 'ESC R' im EEPROM abspeichern

-> Das erwähnte MiniTerminal-Testprogramm beinhaltet diese Funktionen.

Speicherformat : 'FT0'+Chr\$(0)+ 'FT1'+Chr\$(0)+ 'FT2'+Chr\$(0) 'FTn'+Chr\$(0)+Chr\$(0)
(im EEPROM)

Festtext ausgeben :

ESC D nr	Festtext mit Nummer 'nr' anzeigen ('nr' von 0 .. 127)
-----------------	---

Beispiel: **PRINT #1,CHR\$(27);"D";CHR\$(2);**

Das MiniTerminal löscht das Display und zeigt den 3. Festtext an.

Dieser Befehl überschreibt die Einstellung am Stecker J4.

Ändert sich der Wert am Stecker J4, wird wieder der Festtext, der dem Wert am Stecker J4 zugeordnet ist, angezeigt. Das bedeutet eine Festtextnummer wird immer dann angewählt, wenn entweder über die serielle Schnittstelle eine Nummer vorgegeben wird oder sich am Stecker J4 die Einstellung ändert (FT-Modus).

Hinweis: Die Festtext-Befehle können unabhängig vom Wert des Pins 'FT/Bus' verwendet werden. D.h. über die serielle Schnittstelle können in **beiden** Betriebsarten ('autom. FT' oder 'Bus') Festtexte angewählt werden.

7.4 Transparent Mode

Im Transparent-Mode hat der Host direkten Zugriff auf den Controller-Chip des Displays (HD44780 oder ähnlich).

ESC & T | in Transparent-Mode wechseln

x0 data	Write CMD (Adresse 0)
x1 data	Write Data (Adresse 1)
x2	Read CMD
x3	Read Data
x4 data	Controller auswählen (nur 40x4)
	data = x0 -> Controller 1
	data = x1 -> Controller 2
sonst	Transparent-Mode verlassen

data Zeichen in Hex
x don't care (die oberen 4 Bits sind ohne Bedeutung)

Beispiel: **PRINT #1,CHR\$(27);"&T";** ' Enter Transparent Mode
 PRINT #1,CHR\$(0);CHR\$(2); ' 2 in Controller-Adresse 0 schreiben
 PRINT #1,CHR\$(255); ' Transparent-Mode verlassen

8 Anschlussbelegungen

J3 Matrix-Tastatur

Zuordnung der Zeilen zu Pins		Zuordnung der Spalten zu Pins	
Spalte 0 ->	2	1	
Spalte 1 ->	4	3	
Spalte 2 ->	6	5	
Spalte 3 ->	8	7	
Spalte 4 ->	10	9	<- Zeile0
Spalte 5 ->	12	11	<- Zeile1
Spalte 6 ->	14	13	<- Zeile2
Spalte 7 ->	16	15	<- Zeile3

Pin nr an Stecker J3

J4 Festtexte (FT-Modus)

Pin	FT-Modus	Beschreibung
1	FT0	Nummer des Festtextes der ausgegeben
2	FT1	werden soll
3	FT2	(7 Bit / 0 127 binär codiert)
4	FT3	
5	FT4	
6	FT5	
7	FT6	
8	FT/Bus	automatischer Festtext (1) / Bus (0)
9	Masse	

Adressauswahl (Bus- Modus)

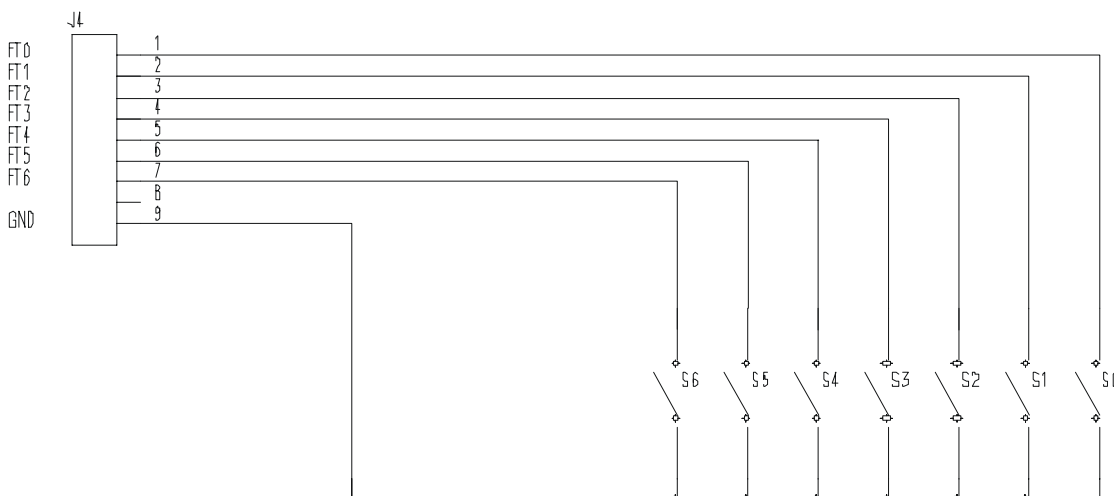
Pin	Bus-Modus	Beschreibung
1	BR1	Binäre Adresscodierung,
2	BR2	Adr. 0....15
3	BR3	
4	BR4	
5		
6		
7		
8	BR5	automatischer Festtext (1) / Bus (0)
9	Masse	

Bitte beachten:

BR1....BR5 sind identisch mit J4 Pin 1...4 und Pin 8.

Es muss gewährleistet sein, dass nie unterschiedliche Potenziale an den o.g. Kontakten anliegen, die somit zu Kurzschlüssen führen !

Anschlussbeispiel für Festtextabfrage über J4



J2 / J1 Anschlußbelegung von LCD-Anzeigen

Pin	Bedeutung	Beschreibung	Pin	Bedeutung	
1	Vss	GND,0V	8	DB1	Data bus
2	VDD	+ 5V	9	DB2	-
3	VEE	for LCD Drive	10	DB3	-
4	RS	Function Select	11	DB4	-
5	R/W	Read/Write	12	DB5	-
6	E	Enable Signal	13	DB6	-
7	DB0	Data Bus Line	14	DB7	-

Belegung gilt nicht für LCD mit 40x4 Zeichen

J1 Anschlußbelegung LCD-Display 40x4

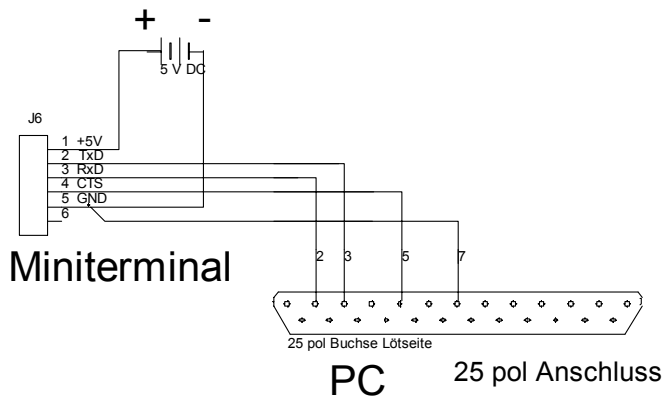
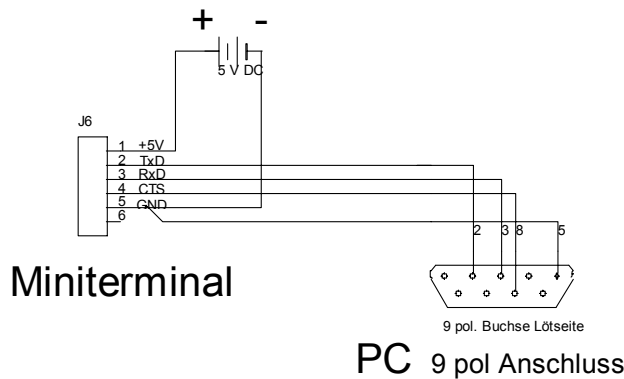
Pin	Bedeutung	Beschreibung	Pin	Bedeutung	Beschreibung
1	DB7	Data bus	9	E	Enable Signal
2	DB6	-	10	R/W	Read/Write
3	DB5	-	11	RS	Function Select
4	DB4	-	12	VEE	for LCD Drive
5	DB3	-	13	Vss	GND,0V
6	DB2	-	14	VDD	+ 5V
7	DB1	-	15	E2	Enable Signal 2
8	DB0	-	16	(NC)	

Bitte beachten: Werden Anschlußkabel für LCD's länger als 5 cm verwendet, müssen Pull-Up-Widerstände an allen Steuer- und Datenleitungen (Pin4 - Pin14) angebracht werden.

J6 Host / Versorgungsspannung

Pin	Bedeutung	Beschreibung
1	+ 5 V	Versorgungsspannung
2	TxD	V24 Ausgang
3	RxD	V24 Eingang
4	CTS	
5	Masse	
6	nicht verwendet	

Kabel MiniTerminal <--> PC

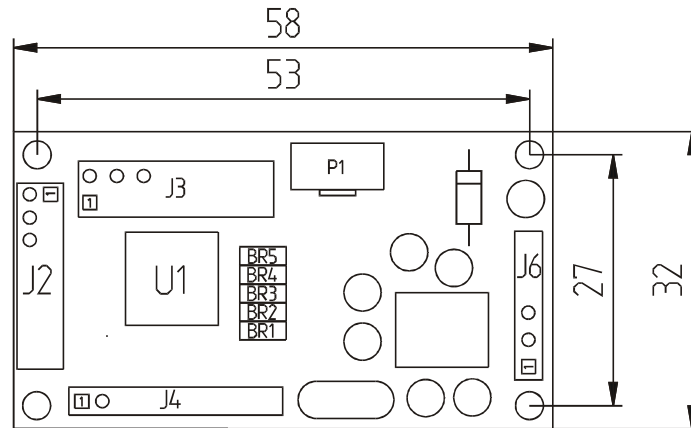


9 Abmessungen der Controller

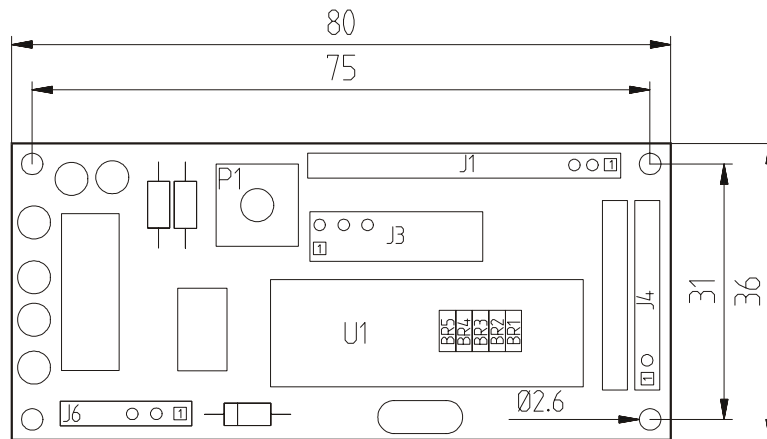
Hinweis : In allen Abbildungen wird der Pin 1 durch eine "1" auf der entsprechenden Seite

des Steckers gekennzeichnet.

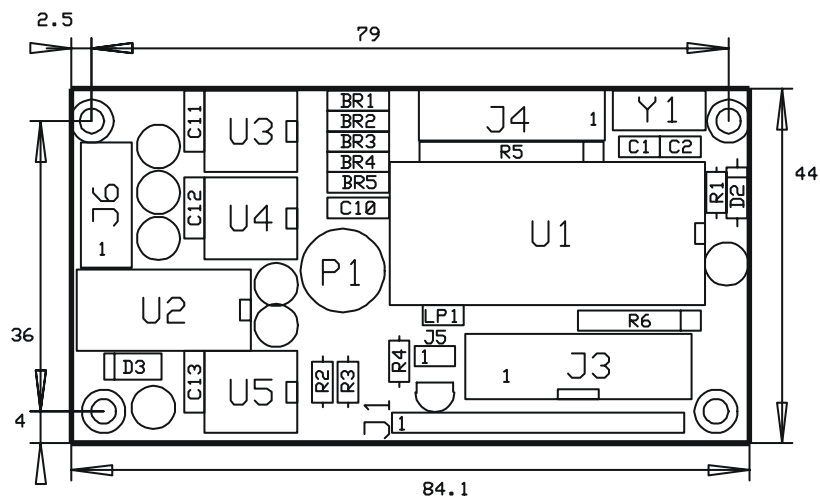
9.1 MiniTerminal für 8 x 2 LCD



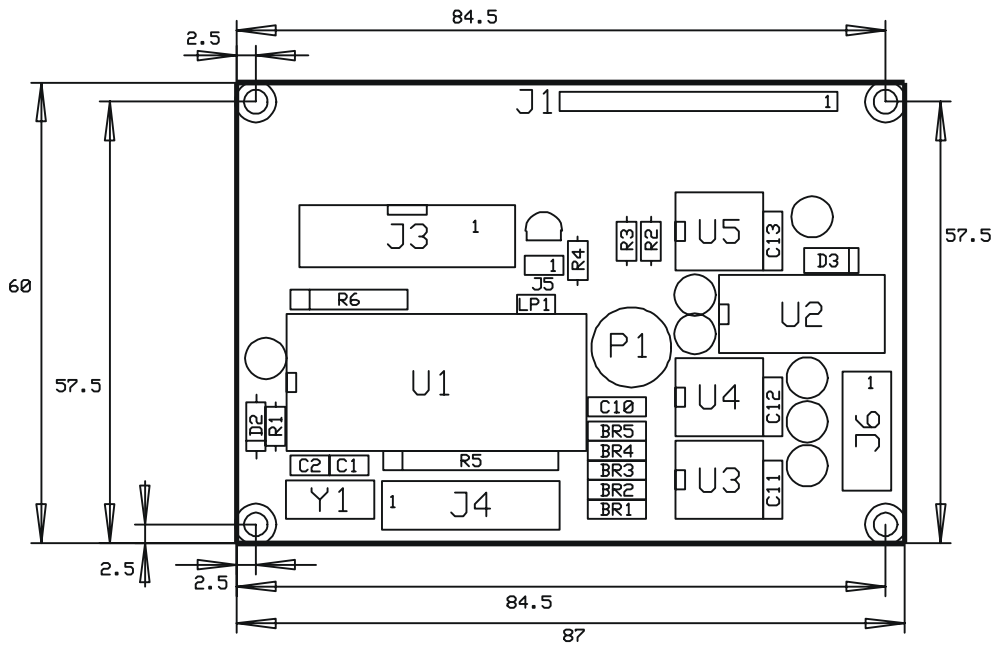
9.2 MiniTerminal für 16 x 1 LCD



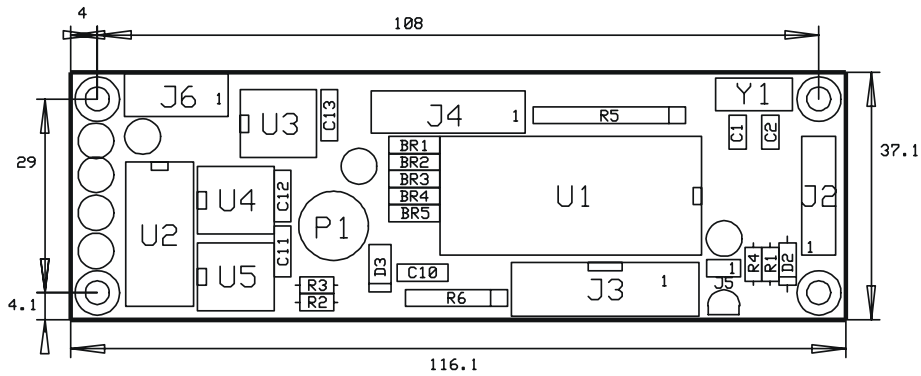
9.3 MiniTerminal für 16 x 2 LCD



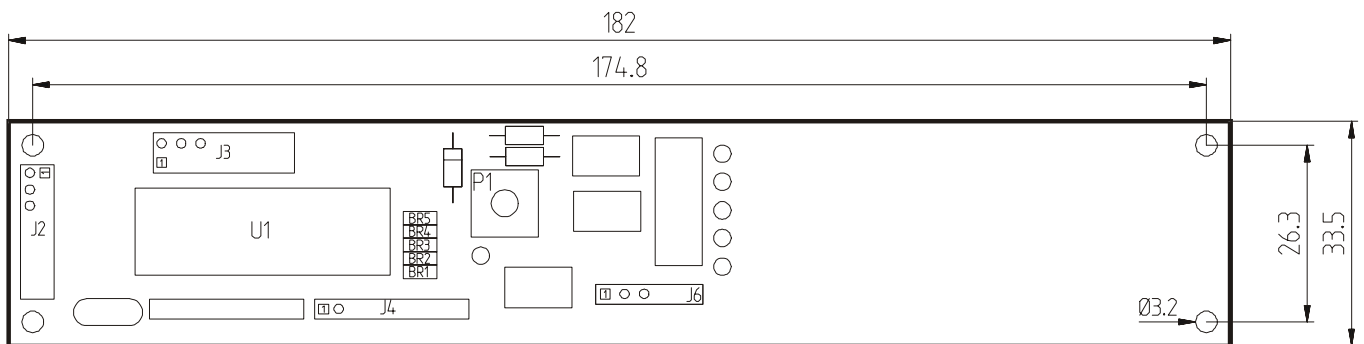
9.4 MiniTerminal für 16 x 4, bzw. 20 x 4 LCD



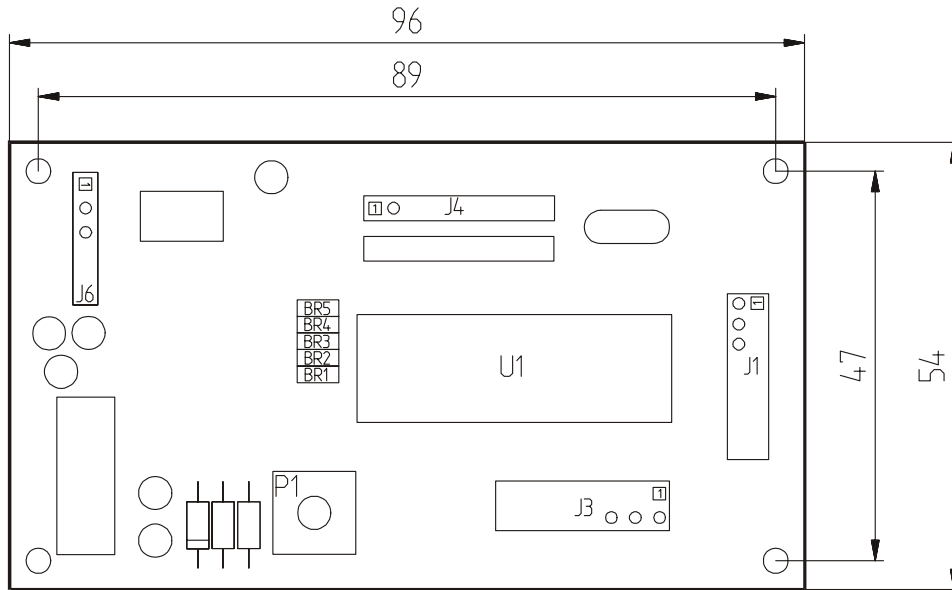
9.5 MiniTerminal für 20 x 2 LCD



9.6 MiniTerminal für 40 x 2 LCD



9.7 MiniTerminal für 40 x 4 LCD



10 Demo Programm für Q-Basic

```
10 ' Demo-Programm für die Festtextanzeige des MiniTerminals
20 '
30 OPEN "COM1:9600,N,8,2,cs0,ds0,cd0,rs" AS 1
40 PRINT
50 PRINT "1 - Festtexte in das MiniTerminal laden"
60 PRINT "2 - Festtexte auf dem PC anzeigen"
70 PRINT "3 - Festtexte auf dem MiniTerminal anzeigen"
80 PRINT "4 - Text auf dem Miniterminal anzeigen"
90 PRINT "5 - Ende  "
100 INPUT CMD
110 ON CMD GOSUB 140,350,490,560,670
120 GOTO 40
130 '-----
140 PRINT "Festtexte in das MiniTerminal laden"
150 PRINT "Ein Festtext wird mit 'ENTER' abgeschlossen"
160 PRINT "Enthält ein Festtext keine Zeichen mehr, "
170 PRINT "wird die Übertragung beendet"
180 PRINT #1,CHR$(27);"R"; : ' Kommando an MiniTerminal schicken
190 INPUT "Text eingeben: ";TEXT$
200 L = LEN(TEXT$)
210 IF L=0 THEN GOTO 310 : ' Übertragung beenden
220 FOR I=1 TO L
230   PRINT #1,MID$(TEXT$,I,1);
240   R$ = INPUT$(1,#1)
250   IF R$ <> MID$(TEXT$,I,1) THEN RETURN : ' Fehler beim Speichern
260 NEXT I
270 PRINT #1,CHR$(0); : ' Text-Ende-Zeichen
280 R$ = INPUT$(1,#1) : ' 0 lesen
290 GOTO 190
300 '
310 PRINT #1,CHR$(0); : ' Text-Ende-Zeichen
320 R$ = INPUT$(1,#1) : ' 0 lesen
330 RETURN
340 '-----
350 PRINT "Im MiniTerminal gespeicherte Texte auf dem PC anzeigen"
360 PRINT
370 PRINT #1,CHR$(27);"W"; : ' Kommando an MiniTerminal schicken
380 NULLFLAG=0
390 A$ = INPUT$(1,#1)
400 IF ASC(A$)=0 GOTO 440
410 NULLFLAG=0
420 PRINT A$;
430 GOTO 390
440 IF NULLFLAG=1 THEN RETURN ' 2. Null --> Ende der Übertragung
450 NULLFLAG=1
460 PRINT
470 GOTO 390
480 '-----
490 PRINT "Fest-Text auf dem MiniTerminal anzeigen"
500 PRINT "Nummer (999 = Ende): ";
510 INPUT NUMMER
520 IF NUMMER=999 THEN RETURN
530 PRINT #1,CHR$(27);"D";CHR$(NUMMER);
540 GOTO 500
550 '-----
560 PRINT "Texte auf dem MiniTerminal ausgeben"
570 PRINT "Der Text wird nach der Eingabe von 'ENTER' ausgegeben"
580 PRINT "Enthält ein Text keine Zeichen mehr, wird die Funktion beendet"
590 INPUT "Text eingeben: ";TEXT$
600 L = LEN(TEXT$)
610 IF L=0 THEN RETURN : ' Übertragung beenden
620 FOR I=1 TO L
630   PRINT #1,MID$(TEXT$,I,1);
640 NEXT I
650 GOTO 590
670 END : ' Programm-Ende
```

11 Character-Font-Tabelle : (darstellbare Zeichen im Hex-Code)

		Higher 4 bit (D4 to D7) of Character Code (Hexadecimal)																			
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F				
Lower 4 bit (D0 to D3) Character Code (Hexadecimal)	0	CG RAM (1)			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	1	CG RAM (2)			!	1	A	Q	a	9			a	7	#	4	a	a	g		
	2	CG RAM (3)			"	2	B	R	b	r			"	4	U	x	p	e			
	3	CG RAM (4)			#	3	C	S	c	s			#	7	T	F	e	a			
	4	CG RAM (5)			\$	4	D	T	d	t			\$	7	T	t	p	a			
	5	CG RAM (6)			%	5	E	U	e	u			%	7	U	u	o	u			
	6	CG RAM (7)			&	6	F	V	f	v			&	7	V	v	a	p	e		
	7	CG RAM (8)			'	7	G	W	w				'	7	W	w	g	w			
	8	CG RAM (1)			(8	H	X	x				(7	X	x	j	x			
	9	CG RAM (2))	9	I	Y	y)	7	Y	y	'	y			
	A	CG RAM (3)			*	A	J	Z	z				*	A	Z	z	j	z			
	B	CG RAM (4)			+	B	K	C	c				+	B	C	c	'	c			
	C	CG RAM (5)			,	C	L	#	1	1			,	C	#	1	1	'	1		
	D	CG RAM (6)			-	D	=	M	1	n)		-	D	=	M	1	n	'	n	
	E	CG RAM (7)			.	E	>	N	^	n	^		.	E	>	N	^	n	'	n	
	F	CG RAM (8)			/	F	?	0	_	o	e		/	F	?	0	_	o	'	e	

12 Software

(A) Kostenloses QBASIC-Programm

Voraussetzungen

- ◆ RS 232 Schnittstelle (COM1 o. 2)
- ◆ EGA oder VGA-Grafikfähig
- ◆ DOS 5.0 oder höher
- ◆ Internetzugang: Das Programm ist ausschließlich über die Homepage www.bue.de zu beziehen.

Diese Programm ist ein im Vergleich zum abgedruckten Demoprogramm stark erweitertes Testprogramm in Qbasic-Quelltext.

Es bietet einem Anwender die einfache Möglichkeit den Funktionsumfang des MiniTerminal-Controllers kennenzulernen und gleich auch anzuwenden.

Es bietet eine einfache Möglichkeit Festtext abzuspeichern, zu erweitern oder zu ändern.

(B) MITAS

Die MiniTerminalAnsteuerSoftware zur einfachen Konfiguration des MiniTerminals.

Dieses Programm bietet dem Anwender die einfache Möglichkeit den Funktionsumfang des **MiniTerminals** kennenzulernen und anzuwenden.

Es bietet vor allem die Möglichkeit, Festtexte abzuspeichern, zu erweitern oder abzuändern sowie den Zeichensatz zu konfigurieren.

Das Programm ist durch die Menübedienung selbsterklärend.

Systemvoraussetzungen:

- Win 3.xx
- Win 95 oder höher
- Win NT 4.0 oder höher

Starten von Diskette:

- Mit Win3.xx Dateimanager bzw. Win95/98/NT Explorer das Laufwerk 'A:\' anwählen
- Verzeichnis '**Windows**' öffnen und Programm '**Terminal.exe**' starten